Presented at Imaging Spectrometry VI (SPIE Vol. 4132), International Symposium on Optical Science and Technology, San Diego, CA, July 30 – August 4, 2000

Advanced Processing for High-bandwidth Sensor Systems

John J. Szymanski,* P. Blain, J. Bloch, C. Brislawn, S. Brumby, M. Caffrey, M. Dunham, J. Frigo, M. Gokhale, N. Harvey, G. Kenyon, W. Kim, J. Layne, D. Lavenier, K. McCabe, M. Mitchell, K. Moore, S. Perkins, R. Porter, S. Robinson, A. Salazar, J. Theiler and A.C. Young

Los Alamos National Laboratory,
Mail Stop C323, Los Alamos, NM 87545

ABSTRACT

Compute performance and algorithm design are key problems of image processing and scientific computing in general. For example, imaging spectrometers are capable of producing data in hundreds of spectral bands with millions of pixels. These data sets show great promise for remote sensing applications, but require new and computationally intensive processing. The goal of the Deployable Adaptive Processing Systems (DAPS) project at Los Alamos National Laboratory is to develop advanced processing hardware and algorithms for high-bandwidth sensor applications. The project has produced electronics for processing multi- and hyper-spectral sensor data, as well as LIDAR data, while employing processing elements using a variety of technologies. The project team is currently working on reconfigurable computing technology and advanced feature extraction techniques, with an emphasis on their application to image and RF signal processing. This paper presents reconfigurable computing technology and advanced feature extraction algorithm work and their application to multi- and hyperspectral image processing. Related projects on genetic algorithms as applied to image processing will be introduced, as will the collaboration between the DAPS project and the DARPA Adaptive Computing Systems program. Further details are presented in other talks during this conference and in other conferences taking place during this symposium.

Key words: multispectral analysis, hyperspectral analysis, image processing, reconfigurable computing, remote sensing

1. INTRODUCTION AND GOALS

The Deployable Adaptive Processing Systems (DAPS) project at Los Alamos National Laboratory (LANL) was formed to apply advanced processing technologies to deal with large amounts of sensor data, typically taken at high data rates. The processing systems are used in both real time and archival (data-center-oriented) applications and are often designed to detect rare signals within these large data sets. The project combines the expertise of algorithm design experts, hardware engineers, and system software engineers into a team that is capable of producing end-to-end processing solutions. The project team members also work on various sensor projects, thereby connecting the DAPS goals to specific application areas. The DAPS team is also collaborating with industry, universities, the Defense Advanced Research Projects Agency (DARPA), and other US Government agencies to have access to and develop future state-of-the-art processing solutions.

In addition to processing system hardware and software design, the team is developing advanced feature-extraction systems for use in sensor applications. These software systems are used with archival data, not for real-time analysis, but algorithms derived from this work might well be applied in real-time applications in the future. For example, the Rapid Feature Identification Project, or RFIP, was started in 1998 for the purpose of applying genetic programming techniques to feature extraction for remote sensing imagery. This project will be introduced in section 5.1 and is also the subject of papers by Perkins et al[1] and Harvey et al[2] at this symposium.

The DAPS project is presently focussing its hardware research efforts on reconfigurable computing (RCC). Hardware solutions used by DAPS span the space of techniques available in the signal/image processing areas, but at the present time RCC is capable of uniquely filling certain processing niches. Other processing elements that might be used by DAPS include digital signal processors (DSPs), general-purpose microprocessors, processor-in-memory chips, and application-specific integrated circuits. Indeed, RCC technology is generally used in conjunction with other technologies, as indicated in Fig. 1.

*Correspondence: Email: szymanski@lanl.gov; Telephone: 505 665 9371; Fax: 505 667 3815

2. TECHNOLOGIES EMPLOYED

The systems produced by the DAPS team use a myriad of hardware technologies, with the typical system having several types of processing elements. As already mentioned, the systems use Digital Signal Processors, general-purpose microprocessors, application-specific integrated circuits and reconfigurable computing (RCC) processing elements. Future systems might include processor-in-memory chips, which would be done in collaboration with the University of Southern California's Information Sciences Institute – East.[3] As RCC is at present the core DAPS processing technology, the rest of this section will concentrate on a description of RCC and the research the DAPS team is pursuing in this area.[4]

2.1 Introduction to Reconfigurable Computing Technology

RCC uses high-gate-density field-programmable gate arrays (FPGAs) as general purpose processing elements. FPGAs used in this manner allow the designer to customize the processing hardware to the task at hand. Resulting performance gains of 10 to 100 or more over conventional microprocessors have been demonstrated.[5,6,7,8]

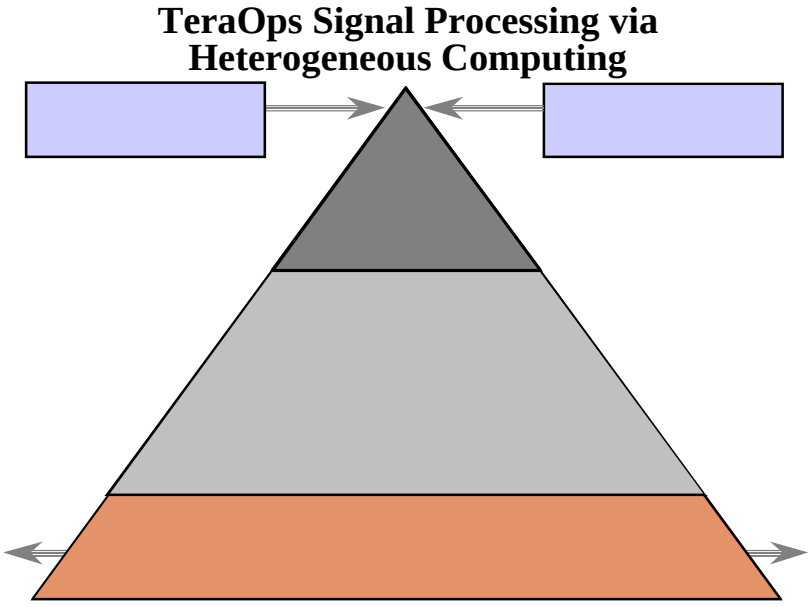

Figure 1. Reconfigurable computing is used in conjunction with other processing technologies.

The FPGA has been on the market for more than a decade; its principal use is for discrete logic integration and board level integrated circuit (IC) count reduction. It is itself a digital IC and represents the highest transistor count of all, up to 100 million transistors[9]. For comparison, the Intel Pentium III is in the neighborhood of 9.5 million transistors[10]. Recent advances yielding this density and improving speed performance have made these parts appropriate for hardware DSP. An historic concern for DSPs has been technology obsolescence. The FPGA market is not driven by DSP applications, so this technology will not lose technical support or future product development as many custom DSP ASICs have in the past.

RCC accelerates applications by exploiting fine-grained parallelism in a given algorithm. A typical FPGA clock speed is an order-of-magnitude slower than typical microprocessors; thus any acceleration is accomplished with parallel execution. In its simplest form, the goal is to have tens, hundreds or more arithmetic computations operating simultaneously. This parallelism puts the FPGA in a unique computational position of being able to maintain throughput while increasing the number of operations. For instance, a microprocessor processes fewer samples in time as filter length increases, while an FPGA exhibits increased logic utilization, but maintains total data throughput by increasing the number of multiply-accumulate units in the design.

Disadvantages of RCC implementations include limited bit precision of operations, fixed-point arithmetic and the need to

program RCC processors with either a hardware description language or with specialized extended high-level languages that contain calls to RCC implementations of core processing routines. The first two limitations require algorithm designers to truly understand the precision needs of the algorithm and how fixed-point arithmetic will impact the stability and precision of results. This challenge is frequently overlooked when designing for microprocessors that deliver 32-bit floating-point performance, but in an FPGA 32-bit floating-point is exceedingly inefficient. Computing techniques also have a large impact on efficiency and performance for FPGAs. For instance, when taking the magnitude of a complex number, the obvious approach uses the square root operation, but the square root operation is inefficient in the hardware implementation. The CORDIC (COordinate Rotation DIgital Computer) algorithm for coordinate transformation is far more efficient in hardware for performing this computation. Designing processing systems in an FPGA often requires creative solutions and understanding of algorithms originally developed, in some cases, 30 years ago. It is well worth the effort, though, because the computation available for given weight, space and power is so much greater that it is feasible to do substantial in situ processing of remotely sensed data.

The DARPA Information Technology Office Adaptive Computing Systems effort had the goal of increasing RCC ease-of-use as well as to introduce more advanced FPGA and board architectures.[11] Elements of this effort have since been integrated into the DARPA Tactical Technology Office (TTO) program. The DAPS team collaborates with these DARPA programs on RCC hardware implementation through the System Level Applications of Adaptive Computing (SLAAC) project, which is producing RCC hardware based on the Xilinx 4000 series FPGAs.[12] A new design is under test for an upgraded SLAAC RCC board using Xilinx Virtex 1000 parts.[13] The role of DAPS is to insert DARPA-developed reconfigurable computing technology into on going LANL applications and to provide challenge problems for the larger Adaptive Computing Systems community. The DAPS team is also working under DARPA auspices on the Streams-C compiler.[14] The compiler synthesizes hardware circuits for multiple FPGAs as well as a multi-threaded software program for the control processor. In comparing compiler-generated hardware circuits against hand-coded VHDL for image processing applications, we have found the compiler generated circuits to be 3 times the area and 1/2 the clock rate of hand-coded designs. The Streams-C program was developed in a few days, while the VHDL version took three months, giving an order of magnitude improvement in productivity. The compiler was applied to the pixel purity index algorithm as part of the ASAPP project described in section 4.3 below.

2.2 DAPS Reconfigurable Computer Architecture II Board

The DAPS project, in collaboration with LANL RF sensor projects, designed and fabricated an RCC board called the Reconfigurable Computer Architecture II (RCA2).[15] It uses three Altera 10k250 Complex Programmable Logic Devices (the CPLD is an industry alternate name of the FPGA) as processing elements organized in a ring architecture. Each of these processors has 40 Kbits of on-chip memory and provides a nominal 250,000 configurable logic gates.[16] Each processor element has 3 independent banks of high performance Zero Bus Turnaround (ZBT) synchronous SRAM that can store intermediate calculations, results, lookup tables, etc. In addition to the individual banks of SRAM, there is a bank of shared SRAM between each of the processors. This memory can be switched between the processors in a single clock cycle. One processor can write data into its shared memory, then change the switch, followed by the next processor simply reading the memory with flipped address lines to automatically transpose the image for the next calculation. The board has flexible, high-speed, on-board communications that enable data to be sent between the memory and the processor elements as well as from one processor to another. The physical form factor is a ‘C’ sized VXIbus single-width card.

There are three high-speed data ports on the front panel of the processor module. Each of the data ports is implemented as a daughtercard so the interface can be configured for input or output and the port can be customized for different sensors. This allows for computing on two input data streams or producing two sets of output results from one input data stream; this is a common data artifact of operations like cross-correlation. The board is designed to handle data rates of at least 100 MB/s, which requires that data be transferred through the high-speed data ports rather than the backplane. The board has flexible, high-speed, on-board communications that enable data to be sent between the memory and the processor elements as well as from one processor to another.

As in most areas of integrated-circuit technology, FPGA development continues. In particular, the Xilinx Virtex series of FPGAs have many features of great interest to the RCC community. Thus, a new version of the RCA-2, the RCA-3, has been developed by Catalina Research, Inc. through a Coorperative Research and Developmet Agreement with LANL. The RCA-3 retains the memory and processor architecture and daughtercard data ports. Changes in the RCA-3 include up to three Xilinx

Virtex XCV1000 series FPGAs in place of the 10k250 processors and a single slot, 6U VME64x form factor.[17]

3. APPLICATION AREAS

DAPS technology is being used in several sensor processing areas. Major application areas include wideband radio frequency signal processing and remotely sensed image processing. The remotely sensed image processing falls under the category of multi-dimensional image processing, as the recent emphasis in DAPS has been on multi- and hyper-spectral, as well as time-series images. Within these general application areas DAPS typically provides solutions for several processing needs and is involved with multiple sensor projects. This section will contain an introduction to the RF applications of DAPS technology. Because of the subject matter of this conference, DAPS work in the processing of remotely sensed imagery will be emphasized for the balance of this paper.

In wideband RF applications, which is loosely defined as searching for signals from 1 – 1000 MHz, DAPS technology is being applied to compression and to accelerating Gabor transformations for signal representation. The compression algorithm is based on subband coding schemes and involves a multirate filter bank, adaptive uniform scalar quantization, and Huffman entropy coding.[18] The compression algorithm is presently running in software, and is being adapted to use on RCC hardware. The Gabor expansion is used to represent signals whose frequency is dependent on time, such as chirps and short duration signals.[19] It is particularly useful for detecting bandpass signals, as it is not invariant with frequency scale (as opposed to wavelet transforms that are scale-invariant). Reconfigurable computing technology is being applied to both the compression and Gabor transformation algorithms.

DAPS technology was also used on the FORTE (Fast On-orbit Recording of Transient Events) RF satellite[20] for an event classifier.[21] FORTE is used to measure and classify electromagnetic transients and associated optical signatures from space. The event classifier consists of a single Texas Instruments TMS320C30 DSP, a radiation-hardened Intel 80C31 microprocessor, and Actel programmable gate array and various types of memory chips. The event classifier is a data filter used to classify events from memory that have been detected by other on-board trigger electronics. The classified events were all downlinked, where subsequent analysis confirmed the accuracy of the event classification. The system was not used to block the downlinking of data, but is a useful, space-based test of this concept for impulsive RF applications.

4. REMOTE SENSING APPLICATIONS

Los Alamos National Laboratory has been working on remote sensing imaging systems for approximately the last decade and has been working on spaceborne astrophysical imaging and particle detection systems for several decades. The emphasis has been on multispectral, hyperspectral and LIDAR remote sensing, with particular expertise in system calibration, data processing, algorithm development and modeling.

4.1 Smoke Plume Detection

The DAPS team has done smoke plume detection both in software and hardware, with the ultimate goal of a real-time plume tracking system. Smoke plumes are difficult to detect and track because they tend to be optically thin, their borders are ill-defined and they change shape with time. Thus, good plume motion estimates are essential and the best results are obtained by considering spatial image segmentation and motion estimation simultaneously. For these reasons, plus the generally large number of unknown inputs, plume modeling is unlikely to help in plume detection. Techniques pursued for plume detection/motion estimation include tradition temporal filtering and optical flow.[22] Most recently, the team produced a plume detection algorithm that combines block matching (such as used in video codecs) with temporal clustering.[23] Two DSP-based hardware systems for plume tracking were constructed as well.

4.2 LIDAR Applications

The DAPS team has worked for many years on LIDAR applications. One system of note is a laser ranging and digital integration system. Because differential absorption LIDAR systems require detection of laser return energy, the range to target must be established. The use of a mobile sensing platform makes this difficult because the roundtrip time for the light pulse is variable and unknown. This information can be extracted from the data in real time by averaging many returns to increase signal-to-noise ratio, then peak detecting to determine target range. To determine the range to the necessary precision requires processing 80 MB/s, far beyond microprocessor or DSP performance limitations. The DAPS team used RCC to solve this problem. The system also digitally integrates the detector data so that integration time is precisely matched

to the pulse width.

4.3 Spectral Processing Applications

DAPS has produced implementations of three general-purpose spectral processing algorithms. The first work was done on the RCA-2 board and was reported by Caffrey et al.[24] The algorithm was a simple dot product operator that represents the application of a matched-filter to a spectral data cube. The data cube had 16 spectral channels in a 512x512 scene. Twelve spectral signatures were applied to the data cube, which required 2.0 s (averaged over multiple trials) on a 400 MHz Pentium II. The RCA-2 implementation required 0.12 s, for a factor of 17 speedup. This design took a portion of a single RCC processor. If the full RCA-2 were used, 32 spectral signatures could be run with no reduction in data rates, resulting in a factor of 45 speedup. Even for a simple algorithm, this speedup illustrates the scalable nature of RCC. Furthermore, the architecture can be scaled at the module level, with multiple modules running in parallel and processing a large number of spectral signatures.

The Accelerating Segmentation And Pixel Purity (ASAPP) project has DAPS team members applying RCC to two spectral algorithms: k-means clustering and end member selection using the Pixel-Purity Index (PPI)[25,26] algorithm presently implemented in the ENVI (ENvironment for Visualizing Images) (™Better Solutions Consulting, LLC) image-processing package. These algorithms were chosen for RCC acceleration because they are compute-intensive and are generally useful for both multispectral and hyperspectral data analysis (k-means clustering) or are a key technique for hyperspectral analysis (PPI). Faster run times also mean the algorithms can be applied to more data sets and can be applied multiple times to the same data set, with different parameters. The user is thus enabled to explore the behavior of the algorithm under different conditions. Papers by Theiler et al.[27,28] in this volume give more details on the project.

The k-means clustering algorithm has many variants, including versions with spatial correlations included[29], different distance metrics, different initialization schemes, etc. Several of these variants have been studied in simulations by Leeser et al.[30] One version of the k-means algorithm is being implemented on Annapolis microsystems WildForce and WildStar RCC boards. The results are discussed in the paper by Theiler et al.[28]

There are a number of algorithms that find end members in hyperspectral images. The PPI was chosen for RCC implementation because it is part of the popular spectral-processing package ENVI and because it is an intuitive algorithm with many potential variants to be considered for hardware acceleration. Theiler et al. discuss this algorithm and its implementation.[27]

5. ADVANCED ALGORITHM DEVELOPMENT

The DAPS team researches advanced image-processing algorithms in support of the same application areas that receive hardware support. Algorithm development is often done to optimize hardware implementations or to evaluate the effects of hardware limitations. For example, algorithms might need to be changed to optimize hardware implementation, and it is important to understand the implications of these changes to the algorithm performance. Furthermore, it is possible that hardware limitations will point to completely different algorithms for a given task than the “standard”. Old algorithms that are not optimal for implementation on a general-purpose processor might be very efficient in a hardware implementation.

5.1 The Rapid Feature Identification Project

A separate algorithm development thrust is represented by the Rapid Feature Identification Project (RFIP), which has the major goal of greatly improving analyst productivity by providing an easy-to-use feature-extraction tool. Perkins et al[1] and Harvey et al[2] discuss the project in more detail in papers at this symposium. Previous publications on the project include papers by Brumby et al[31,32], Theiler et al[33] and Harvey et al[34].

Traditional feature extraction algorithms often relay on a given geometry and set of observing conditions. Steps are taken to correct or compensate for changes in these conditions, meeting with varying levels of success. The remote sensing researchers at LANL, including some DAPS team members, follow similar approaches, with a particular emphasis on including atmospheric and sensor physics in the algorithms. Clearly, if such a standard approach works well, then it is the preferred method. However, often a researcher is faced with trying many different possible image-processing steps in various combinations and with a plethora of input parameters. Because computers are better than humans at trial-and-error solutions, the RFIP software uses a different strategy. Rather than design feature-extraction tools by hand, the RFIP exploits recent

advances in machine learning methods to develop such tools semi-automatically, using only a small collection of hand-classified images as a starting point. The RFIP tools make it possible for an analyst to produce a feature-extraction tool suitable for a given set of observation conditions, tuned for a specific feature.

The RFIP uses a genetic programming technique to evolve extraction tools for features of interest. Genetic programming is a search technique inspired by genetics: analogies of mutation and the exchange of genetic material are used to move a population of solutions through parameter space. The space consists of sequences of primitive image-processing operators and their parameters. The software that evolves the feature-extraction tools is called Genie, for GENetic Image Exploitation.

Training is accomplished with an intuitive interface (called Aladdin; see Fig. 2), where the user identifies features of interest by coloring them green. Areas of the image that do not encompass the feature of interest are colored red. The user interface is another important feature of the system, because it allows analysts to use the system with minimal instruction. Using the training data the genetic programming tool searches a large space of image-processing algorithms, choosing the primitives, their order of execution, their inputs, their parameters and where their results are stored. A fitness function is used to describe the success of a given solution for the problem at hand. The output of the tool is an image-processing pipeline that extracts the feature of interest. The user can iterate and provide further guidance to Genie for subsequent evolutionary cycles.

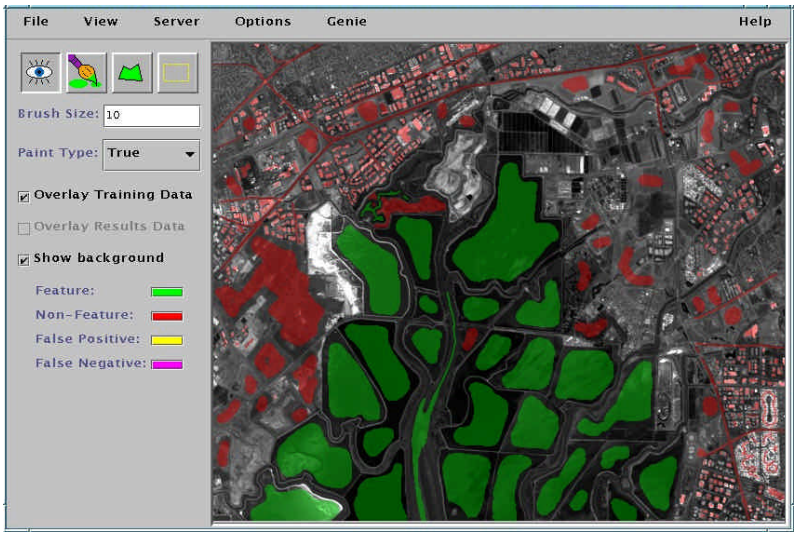

Figure 2. The Aladdin user interface. Features of interest are colored in green and areas to be excluded are colored red (colors not shown above).

Genie is a hybrid system, which produces its final algorithm by first applying the genetic programming technique. This will generally produce several image-processing pipelines, whose outputs are combined with the input data set and are operated on by a conventional linear classification algorithm.[1] The final result of a Genie run is the output of the linear classifier, with the optimal combination of image-processing pipelines.

The Genie/Aladdin system allows an experienced researcher to provide more sophisticated inputs. For example, the output of a well-understood feature-extraction tool can be used as input to Genie. The Genie/Aladdin system is not designed to replace traditional feature-extraction algorithms, but to use them as inputs to seek higher-level answers. Also, the software can use co-registered data sets from different sensor types. This is a capability still in the early stages of development in the remote-sensing science community. Likewise, Genie inherently uses combined spectral/spatial image processing, which is of great promise for future exploitation both by Genie and for hand-designed algorithms.

The present set of software tools does not require special hardware. Future enhancements include running the tools on a network of workstations and accelerating the search with RCC.[35]

5.2 Other Feature-Extraction Techniques

The project team is also researching the use of support-vector machines (SVMs) in feature extraction. SVMs are another approach in the area of machine learning that are supported by more mathematical rigor than genetic algorithms. Following the work of Vapnik[36], a system similar to Genie, but using SVMs, is under development. Aladdin would be the user interface. A further possibility under study is to use Genie in association with an SVM, exploiting the advantages of each approach.[1]

Another area of study is spiking neural networks and their relation to mammalian vision systems. The idea is to apply recent results in mammalian neurobiology to image segmentation. Typically artificial neural networks have ignored the relative

timing of neural impulses to represent information, whereas evidence suggests that the generation of discrete pulse trains provides biological neurons with a natural solution to the problem of dynamic binding and image segmentation. Kenyon et al have achieved excellent results in modeling a retinal circuit that responds to objects as observed in experiment.[37] In particular, the modeled retinal circuit reproduces the temporal pattern of neuron outputs. The retina appears to segment images by synchronizing the output of neurons that are aligned with an object, potentially enabling subsequent processing layers to detect that object. One goal of the research is to apply these retinal circuits to image segmentation of image sequences and multispectral images. One advantage of the retinal circuits is that they are inherently parallel, an obvious advantage for efficient RCC implementation using DAPS technology.

6. SUMMARY

This paper discusses the Deployable Adaptive Processing Systems project and associated projects at the Los Alamos National Laboratory. The project produces high-performance data processing hardware and software for wideband RF sensors and spectral imaging systems. The project members work on hardware design as well as algorithm development, with an emphasis on adaptive computing techniques. Several closely related projects are devoted to research in reconfigurable computing hardware and software tools, genetic programming techniques applied to remote sensing and implementations of specific spectral-processing algorithms on reconfigurable computing platforms.

REFERENCES

1. Perkins, S.J., J. Theiler, S.P. Brumby, N.R. Harvey, R. Porter, J.J. Szymanski, and J.J. Bloch, "GENIE – a hybrid genetic algorithm for feature classification in multi-spectral images," to appear in proceedings of SPIE **4120** (2000).
2. Harvey, N.R. S.P. Brumby, S.J. Perkins, R. Porter, J. Theiler, J.J. Szymanski, and J.J. Bloch, "Competition and collaboration: Parallel evolution of image processing tools for multispectral imagery," to appear in proceedings of SPIE **4132** (2000).
3. http://www.east.isi.edu/SLIIC/
4. http://rcc.lanl.gov/
5. Villasenor, J., B. Schoner,, K. Chia, C. Zapata, H. Kim, C. Jones, S. Lansing, and B. Mangione-Smith, "Configurable computing solutions for automatic target recognition," 4th IEEE Symposium on FPGAs for Custom Computing Machines (FCCM '96) (1996).
6. Villasenor, J. and B. Hutchings, "The Flexibility of Configurable Computing," IEEE Signal Processing Magazine, 67-84 (1998).
7. Villasenor, J. and W. Mangione-Smith, "Configurable Computing," Scientific American, June (1997).
8. Michael Rencher and Brad Hutchins, "Automatic Target Recognition on Splash 2," IEEE Symposium on FPGAs for Custom Computing Machines (FCCM '97) (1997).
9. Approximate for Altera 20K1000.
10. http://www.intel.com/pressroom/kits/processors/quickref.htm
11. http://www.darpa.mil/ito/research/acs/index.html
12. http://www.east.isi.edu/SLAAC/
13. http://www.xilinx.com/products/virtex.htm
14. Gokhale, M.B., J.M. Stone, J. Arnold, and M. Kalinowski, "Stream-oriented FPGA computing in the streams-C high level language," IEEE International Symposium on FPGAs for Custom Computing Machines (FCCM 2000) (2000).
15. Robinson, S. H., Caffrey, M. P., and Dunham, M, E., "Reconfigurable Computer Array: The bridge between high speed sensors and low speed computing," Eighth International Workshop on Field-Programmable Logic and Applications, FPL '98, 159-168 (1998).
16. Altera Data Book, p. 22, (1998).
17. http://www.catalinaresearch.com/CRItechdata/default.htm
18. Brislawn, C.M., S.H. Robinson, and S.A. Crockett, "Subband coding of RF signals in reconfigurable computing hardware," Proc. IEEE MWSCAS, 1135-1138 (1999).

19. Feichtinger H., and T. Strohmer Eds., *Gabor Analysis & Algorithms*, (Birkhauser, Boston, 1998) and R. Crochiere & L. Rabiner, *Multi-Rate Digital Signal Processing*, Chapter 7 (Prentice-Hall, 1983).
20. http://forte.lanl.gov
21. Moore, K.R., P.C. Blain, S.D. Briles, and R.G. Jones, "Classification of RF transients in space using digital signal processing and neural network techniques," proceedings of SPIE **2492**, 995-1006 (1995).
22. Horn, B.K.P., and B. Schunk, "Determining optical flow,'" Artificial Intelligence **17**, 309-354 (1981).
23. Kim, W.H., C. Brislawn, "Plume detection and tracking in video data," Los Alamos National Laboratory technical report LAUR-00-15 (2000).
24. Caffrey, M., J.J. Szymanski, A. Begtrup, J. Layne, T. Nelson, S. Robinson, A. Salazar, J. Theiler, "High performance signal and image processing for remote sensing using reconfigurable computers," proceedings of SPIE **3807**, 142-149 (1999).
25. Boardman, J. W., "Automated spectral unmixing of AVIRIS data using convex geometry concepts," Summaries, Fourth JPL Airborne Geoscience Workshop, JPL Publication **93-26**, **v. 1**, 11-14 (1993).
26. Boardman, J. W., F. A. Kruse, and R. O. Green, "Mapping target signatures via partial unmixing of AVIRIS data," Summaries, Fifth JPL Airborne Earth Science Workshop, JPL Publication **95-1**, **v. 1**, 23-26 (1995).
27. Theiler, J., D.D. Lavenier, N.R. Harvey, S.J. Perkins, and J.J. Szymanski, "Using blocks of skewers for faster computation of pixel purity index," to appear in proceedings of SPIE **4132** (2000).
28. Theiler, J., M. Leeser, M. Estlick, and J.J. Szymanski, "Design issues for hardware implementation of an algorithm for segmenting hyperspectral imagery," to appear in proceedings of SPIE **4132** (2000).
29. Theiler, J., and G. Gisler, "A contiguity-enhanced k-means clustering algorithm for unsupervised multispectral image segmentation," proceedings of SPIE **3159**, 108-118 (1997).
30. Leeser, M., J. Theiler, M. Estlick, and J.J. Szymanski, "Design tradeoffs in a hardware implementation of the k-means clustering algorithm," to appear in proceedings of First IEEE Sensor Array and Multiprocessing Workshop (2000).
31. Brumby, S.P., J. Theiler, S.J. Perkins, N. Harvey, J.J. Szymanski, J.J. Bloch, and M. Mitchell, "Investigation of Image Feature Extraction by a Genetic Algorithm," proceedings of SPIE **3812**, 24-31 (1999).
32. Brumby, S.P., N.R. Harvey, S. Perkins, R.B. Porter, J.J. Szymanski, J. Theiler, and J.J. Bloch, "A genetic algorithm for combining new and existing image processing tools for multispectral imagery," to appear in proceedings of SPIE **4049** (2000).
33. Theiler, J., N.R. Harvey, S.P. Brumby, J.J. Szymanski, S. Alferink, S. Perkins, R. Porter, and J.J. Bloch, "Evolving Retrieval Algorithms with a Genetic Programming Scheme," proceedings of SPIE **3753**, 416-425 (1999).
34. Harvey, N.R., S. Perkins, S.P. Brumby, J. Theiler, R.B. Porter, A.C. Young, A.K. Varghese, J.J. Szymanski, and J.J. Bloch, "Finding Golf Courses: The Ultra High Tech Approach," to appear in the proceedings of EvoIASP 2000: The Second European Workshop on Evolutionary Computation in Image Analysis and Signal Processing (2000).
35. Porter, R., K. McCabe, and N. Bergmann, "An applications approach to evolvable hardware," First NASA/DOD Workshop on Evolvable Hardware (1999).
36. Vapnik, V., *The nature of statistical learning theory*, Springer-Verlag, New York (1995).
37. Kenyon, G.T., K.R. Moore, and D.W. Marshak, "Synchrony of retinal ganglion cells is a more sensitive measure of stimulus intensity than firing rate," in preparation.